

**EXPEDITED PROCESSING
RESPONSE TO OFFICE ACTION****REMARKS**

Applicant had appealed this application to the Board of Patent Appeals and Interferences. However, the Office issued a new Office Action on September 20, 2005 asserting new grounds of rejection of all of the pending claims.

Applicant herein respectfully traverses the new grounds of rejection.

Claims 1-24 are pending in this application. The Office rejected claims 11-13, 16, and 17 as anticipated by U.S. Patent No. 6,076,108 issued to Courts et al. (hereinafter Courts). The Office further rejected claim 14 as obvious over Courts and Prabandham. The Office further rejected claim 1-10, 15, 18, 22, and 23 as obvious over Courts and Prabandham further in view of Ng.

Since Courts is the primary reference for all of the rejections in the latest Office Action and Courts is a newly cited reference, all previous rejections in this application are essentially moot.

Applicant respectfully traverses all of these rejections, as Courts simply does not teach that for which it has been cited.

Background Information

Web sites often divide the tasks of servicing requests into a three tier system with a different server (or plurality of servers) to handle each tier. The first, front-end tier is the http server(s) that processes the http aspects of a transaction. The second tier is the application server(s). The application servers handle the content specific processing for the transactions. The third tier comprises database server(s) that store the data needed to process requests. Within each tier, the Web site server system may have multiple, redundant servers. Particularly, any given server can only service so

**EXPEDITED PROCESSING
RESPONSE TO OFFICE ACTION**

many requests in a given period. If the Web site expects more traffic than a single server can handle, it may maintain multiple servers which can serve the same content. In such situations, since http is a connectionless protocol, one request from a particular client can be directed to one application server while the next request from the same client machine might be directed to a different application server. Accordingly, a means must be provided for the various servers to access the session data developed by another, redundant server.

A common way of enabling such sharing of http session data is by use of a database server that is accessible to the plurality of application servers for storing session data. Particularly, an application server will store session data in local memory, but will also write a copy of the session data to the session database. If a different server services a request from a client, that different server can go to the database and read the session data for the corresponding session.

In the prior art, the session data for a session is updated in both the local memory and the database each time a request causes a change in the data. Writing to the database is a relatively expensive process in terms of processing power and time. Accordingly, the present invention is designed to reduce the number of writes to an http session database in order to conserve system resources.

The Present Invention

The present invention pertains to a method and apparatus for updating a session database that is accessible by multiple servers in a Web environment. In accordance with the invention, each server maintains http session data in a local memory. This

**EXPEDITED PROCESSING
RESPONSE TO OFFICE ACTION**

copy of the http session data will be updated every time there is a change in the session data. The servers automatically write at designated times a copy of the session data to a common database shared by all of the servers. In a preferred embodiment of the invention, the designated time is periodic. In alternate embodiments, the servers may write the session data to the database after a specified number of requests in that session have been received. In another embodiment, the servers may write the session data to the database after a specified number of changes to the session data have been made.

The Courts Reference

The Office asserts that courts teaches a method of maintaining session data in a server system comprising the steps of (1) storing data for each session in a memory local to a server servicing said session; (2) writing a copy of said data for each said session stored in said local memory into a central memory accessible to all servers of said server system at designated times, said designated times being a function of a predetermined time interval since a last write to said database of data for said sessions. The Office asserts that this latter teaching is found in column 9, lines 36-52 and column 2, lines 54-61. Specifically, the Office asserts that Courts discloses a session data stored in a global session server after the web page is built, and the storage of data based on time.

Discussion

Applicant respectfully traverses. Courts teaches exactly that which is disclosed as prior art in the Background of the Invention section of the present application.

EXPEDITED PROCESSING
RESPONSE TO OFFICE ACTION

Exactly as discussed in the Background of the Invention section of the present application, Courts writes the session data to the global session server after every session update. This is exactly the prior art that the present invention improves upon. As noted above, in the present invention, the session data in the central session data store is not updated after every session update but only after a predetermined interval.

Referring to the exact portions of courts to which the office referrer is, they state:

Services 30, trend data collection layer 32 and management console 34 span the four layers 12, 14, 16 and 18. Trend data collection layer 32 can store data, based upon user activity over time, in a trend database 36. One use for trend database 36 is to be mined for data to populate a profile database 38. This profile data can be used to characterize individual users and can be fed back into presentation layer 14 and business layer 16. (Column 2, lines 54-61).

In particular, the state information can be stored as session data 234 created for each user session. Session data 234 can be created when a user initiates a web interaction and there is no current session data for that user. When servicing a request, web system engine 230 can create a session cache 236 and provide session cache 236 with a session ID. Using the session ID, session cache 236 can then request the corresponding session data 234 from global session server 232. If the session data 234 is not already locked, global session server 232 can provide session data 234 to session cache 236 and lock session data 234 from concurrent access. If the session data 234 is already locked, then another session cache 236 is currently processing a request for that user session which needs to be completed. After completion, session data 234 is updated and made available. Once session cache 236 receives the session data 232, web system 230 can use this session information to help build a responsive web page. When the web page is built, it is sent to the requesting user. The session data 234 is then updated, stored in global session server 232, and released for access by another session cache 236. In one implementation, session cache 236 is not released until after one or more shadow copies of session data 234 have been made. (Column 9, lines 36-52).

**EXPEDITED PROCESSING
RESPONSE TO OFFICE ACTION**

Column 2 is general background and is essentially irrelevant to the present issue. Column 9, on the other hand, is exactly on point with the present issue.

Particularly, column 9 quite clearly describes a system entirely consistent with the prior art described in the Background of the Invention section of the present application in which the global session server 232 is updated with the new session state information 234 every time it is changed, and not at "designated times, said designated times being a function of a predetermined time interval since a last write to said database of the data for said sessions". (Claim 11). Specifically note column 9, lines 43-50, underlined above. These lines clearly describe that every time a local session data 234 is updated, it is sent to the global session server 232. Note, in particular, column 9, lines 48-50, which state, "the session data 234 is then updated, stored in global session server 232, and released for access by another session cache 236". It could not be any clearer that Courts does not teach the present invention. Since claims 12, 13, 16, and 17 depend from claim 11, they distinguished over the prior art for at least the same reasons.

The Office rejected dependent claim 14 as obvious over Court in of view of Prabandham. However, Prabandham teaches nothing that was lacking from Courts as discussed above in connection with claim 11, from which claim 14 depends. Accordingly, claim 14 also distinguishes over the prior art of record for at least the reasons set forth above in connection with claim 11.

The Office further rejected claims 1-10, 15, 18, 22, and 23 under 35 U.S.C. 103(a) as unpatentable over Court and Prabandham and further in view of Ng.

**EXPEDITED PROCESSING
RESPONSE TO OFFICE ACTION**

However, once again, independent claim 1 recites “a second computer program adapted to write to said database a copy of said HTTP session data for each said HTTP session at a designated time that is a function of a predetermined time interval since the last write to said database of http session object data for said http session”.

Accordingly, claim 1 distinguishes over the prior art for at least all of the same reasons given above in connection with independent claim 11. Ng does not disclose the teachings lacking from the other references discussed above.

Likewise, independent claim 18 recites “a second computer program adapted to write a copy of said http session data for each said http session in said database at designated times, said designated times determined as a function of at least one of (a) the number of times the http session object data is updated in said local memory and (b) the number of times said http request in said http session is serviced”. Accordingly, claim 18 distinguishes over the prior art for essentially the same reasons as independent claims 1 and 11 as discussed above.

Conclusion

In view of the foregoing amendments and remarks, this application is now in condition for allowance. Applicant respectfully requests the Examiner to issue a Notice of Allowance at the earliest possible date. The Examiner is invited to contact

PATENT

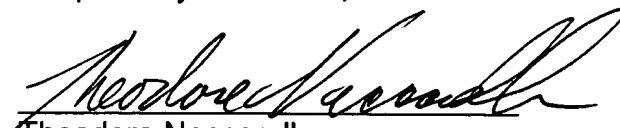
ATTORNEY DOCKET NO. RSW9-2001-0063-US1
Patent Application No. 09/823,120

EXPEDITED PROCESSING
RESPONSE TO OFFICE ACTION

Applicant's undersigned counsel by telephone call in order to further the prosecution of this case in any way.

Respectfully submitted,

Dated: December 20, 2005



Theodore Naccarella
Registration No. 33,023
Synnestvedt & Lechner LLP
2600 Aramark Tower
1101 Market Street
Philadelphia, PA 19107
Telephone: (215) 923-4466
Facsimile: (215) 923-2189

TXN:pmf

M:\ITNACCARELLA\CLIENTS\IBM\24881\PATENT OFFICE\RESPONSE TO 9-20-05 OFFICE ACTION.DOC